Constructing lower bounds for van der Waerden numbers using neural networks

Leonhard Schwarz

TU Berlin

February 2nd 2023

Table of Contents

Van der Waerden numbers

- Color the numbers $1, \ldots, N$ with r different colors
- The van der Waerden number $W(r, k)$ is the smallest N such that each coloring contains a k-term arithmetic progression of the same color
- Example of a 5-term arithmetic progression: $3, 10, 17, 24, 31$
- Existence follows from Van der Waerden's Theorem (1927)

Exact values

Apart from the trivial ones ($k = 1, 2$; $r = 1$), only 7 van der Waerden numbers are known:

- $W(2,3) = 9$
- $W(2,4) = 35$
- $W(2,5) = 178$
- $W(2,6) = 1132$
- $W(3,3) = 27$
- $W(3,4) = 293$
- $W(4,3) = 76$

A general lower bound in the case of 2 colors

- Berlekamp proved that

$$p \cdot 2^p \leq W(2, p+1)$$

  holds for a prime number p (1968)
- Using Betrand's postulate, as well as the monotonicity of the van der Waerden numbers, this implies an exponential lower bound for all 2-color van der Waerden numbers

Specific lower bounds

The following lower bounds are due to Herwig, Heule, van Lambalgen and van Maaren (2007), Rabung and Lotts (2012) and Monroe (2019):

k, r	2	3	4	5	6
3	9	27	76	>170	>223
4	35	293	>1,048	>2,254	>9,778
5	178	>2,173	>17,705	>98,740	>98,748
6	1,132	>11,191	>157,209	>786,740	>1,555,549
7	>3,703	>48,811	>2,284,751	>15,993,257	>111,952,799

Table: Exact values and lower bounds for van der Waerden numbers

All of those lower bounds were achieved by using a method called *Zipping* developed by Herwig et al.

Encoding and scoring function

Consider the 2-color case:

- Each coloring can be encoded as an element of $\{0,1\}^N$
- The score of a coloring C is the lowest index m, such that $C[:m]$ contains a k-term arithmetic progression of the same color

Idea behind the scoring function:

- Loop over elements of C
- Save all indices of a color in a list / set
- When seeing an element of color c at index j, iterate over ealier indices i of color c and check whether there exists a monochromatic k-term arithmetic progression starting at i and ending at j (this can be done with an expected runtime in $O(j)$)

The expected runtime of this is in $O(N^2)$

The r-color case

Ideas for the 2-color case can be extended to the r-color case, but:

- working with $\{0, \ldots, r-1\}$-alphabet seemed to decrease efficiency of the neural network

Encoding the r-color case can also be done via elements of $\{0,1\}^{r \cdot N}$ but:

- this increases the runtime of each generation since larger vectors have to be generated
- this allows for multiple colors per number which slows down progress because the NN has to "learn" to use at least one 1 per r elements while also not using too many 1s as this makes monochromatic progressions more likely

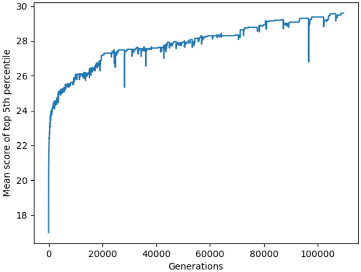
Results for $W(5,3)$

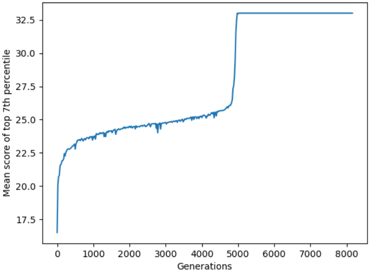
- Current lower bound is 170
- Naive implementation as element of $\{0,1\}^{5\cdot 171}$ seemed to progress faster than using elements from $\{0,1,2,3,4\}^{171}$
- Nevertheless, the main bottleneck was running time since each individual had 855 entries
- Neural Network also had problems with "learning" to not use too many colors
- Best individual achieved score of 17

Results for 2-color case

$W(2,4)$:

- Exact value of $W(2,4)$ is 35
- Neural Network was consistently able to produce examples with score of > 30 after enough iterations but never achieved 34
- Increasing the learning rate led to fast convergence to local maxima
- Naive iterative algorithm can calculate this value within a few seconds





$W(2,5)$:

- Exact value of $W(2,5)$ is 178
- Neural Network was consistently able to produce examples with values around 50 (maximum was 54) but seemed to be stuck there a lot of times
- Decreasing the learning rate led to steady progress but runtime became an issue

$W(2,7)$:

- Current lower bound is 3703
- Since elements of $\{0,1\}^{3704}$ had to be used, the main bottleneck was runtime
- Neural Network produced examples with length 139 and was still progressing

Conclusion

- Wagner's method does not seem to be able to produce any examples close to the unknown van der Waerden numbers with this implementation
- The neural networks also tend to get stuck in local maxima which can be dealt with by altering parameters like the learning rate or the amount of individuals kept in each generation but this slows down progress

References

- Berlekamp, E. (1968). "A construction for partitions which avoid long arithmetic progressions"
- Herwig, P. R.; Heule, M. J. H.; van Lambalgen, P. M.; van Maaren, H. (2007). "A method to construct lower bounds for van der Waerden numbers"
- Monroe, D. (2019). "New Lower Bounds for van der Waerden Numbers Using Distributed Computing"
- Rabung, J.; Lotts, M. (2012). "Improving the use of cyclic zippers in finding lower bounds for van der Waerden numbers"
- Wagner, A.Z (2021). "Constructions in combinatorics via neural networks"